UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
13/595,860	08/27/2012	Rema Ananthanarayanan	IN920090109US2(790.045C)	8033

89885 7590 01/30/2017

FERENCE & ASSOCIATES LLC
409 BROAD STREET
PITTSBURGH, PA 15143

EXAMINER	
TAYLOR, BROOKE JAZMOND	

ART UNIT	PAPER NUMBER
2181	

MAIL DATE	DELIVERY MODE
01/30/2017	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Ex parte REMA ANANTHANARAYANAN, VINATHA CHATURVEDI, VIJIL E. CHENTHAMARAKSHAN, PRASAD M. DESHPANDE, RAGHURAM KRISHNAPURAM, and SHAJEER K. MOHAMMED

Appeal 2016-000574
Application 13/595,860
Technology Center 2100

Before CARLA M. KRIVAK, AMBER L. HAGY, and AARON W. MOORE, *Administrative Patent Judges*.

KRIVAK, *Administrative Patent Judge*.

DECISION ON APPEAL

Appellants appeal under 35 U.S.C. § 134(a) from a Final Rejection of claims 1–10 and 12–14. Claim 11 has been canceled. We have jurisdiction under 35 U.S.C. § 6(b).

We reverse.

STATEMENT OF THE CASE

Appellants' invention is directed to "simplified and automated arrangements and methods for extracting dependency data from packages of bundled products, thus permitting more efficient installation and disk space usage in connection with the bundled product" (Spec. ¶ 7).

Independent claim 1, reproduced below, is exemplary of the subject matter on appeal.

> 1. A method comprising:
> utilizing at least one processor to execute computer code configured to perform the steps of:
> extracting file system structures corresponding to each of at least two software products;
> identifying, from the file system structures, matching subtrees that represent subtrees of more than one directory across the at least two software products; and
> based on the candidate subtrees, generating a dependency graph which indicates:
> common components across at least two software products; and
> dependencies among the common components.

REFERENCES and REJECTIONS

The Examiner rejected claims 1–8, 10, and 13 under 35 U.S.C. § 102(b) as anticipated by Gupta (US 2006/0026157 A1; Feb 2, 2006).

The Examiner rejected claims 9 and 12 under 35 U.S.C. § 103(a) based upon the teachings of Gupta and Appellants' Admitted Prior Art (AAPA).

The Examiner rejected claim 14 under 35 U.S.C. § 103(a) based upon the teachings of Gupta and Wookey (US 2008/0201705 A1).

The Examiner rejected claims 1–8 and 13 under 35 U.S.C. § 103(a) based upon the teachings of Gupta and DeHaan (US 2010/0223609 A1).

The Examiner rejected claims 9, 10, and 12 under 35 U.S.C. § 103(a) based upon the teachings of Gupta, DeHaan, and AAPA.

The Examiner rejected claim 14 under 35 U.S.C. § 103(a) based upon the teachings of Gupta, DeHaan, and Wookey.

ANALYSIS

Rejection under 35 U.S.C. § 102

Appellants contend the Examiner erred in finding Gupta discloses all the elements of Appellants' claims including the limitations "extracting file system structures corresponding to each of at least two software products" and "identifying, from the file system structures, matching subtrees that represent subtrees of more than one directory across the at least two software products" (App. Br. 11–12; Final Act. 3–4; Ans. 3). Particularly, Appellants contend Gupta is directed to HTML pages and items available on the Web, which is not the same as file system structures of software products as claimed (App. Br. 11). Appellants further contend the Document Model Object[1] (DOM) tree disclosed in Gupta does not identify, from file system structures, matching subtrees of more than one directory across at least two software products as claimed. We agree.

[1] "Document Object Model" is a specification "that describes the structure of dynamic HTML and XML documents in a way that allows them to be manipulated through a Web browser." Microsoft Computer Dictionary, Fifth Edition 171 (Microsoft Press 2002).

The Examiner finds Gupta's data source is "equivalent to the software product of [the] instant application. (**Paragraph [0002 and 0008]**)" (Ans. 3; Final Act. 3). However, the Examiner has merely stated this without explanation as to how file system structures[2] in software products are the same as Gupta's data sources.[3] Gupta is directed to extracting generic data from semi-structured documents; in contrast, claim 1 specifically requires "extracting file system structures corresponding to each of at least two software products." As to the Examiner finding the DOM tree, in Figure 3 and paragraphs 41–45 of Gupta, discloses "identifying, from the file system structures, matching subtrees that represent subtrees of more than one directory across the at least two software products" as claimed, we do not agree (Final Act. 3). The Examiner has not shown where in Gupta matching subtrees that represent subtrees of more than one directory across at least two software products are identified.

Thus, we agree with Appellants the Examiner erred in finding Gupta anticipates claims 1–8, 10 and 13.

---

[2] "The file system structure is the most basic level of organization in an operating system. Almost all of the ways an operating system interacts with its users, applications, and security model are dependent upon the way it organizes files on storage devices. Providing a common file system structure ensures users and programs are able to access and write files." Chapter 1, Section 1.1 Copyright © 2006 Red Hat, Inc. https://www.centos.org/docs/5/html/Deployment_Guide-en-US/ch-filesystem.html, last visited Jan. 17, 2017. *See also* http://cs.gmu.edu/~menasce/cs471/slides/ch12.pdf (2003), last visited Jan. 17, 2017.

[3] Web Style Guide: Basic Design Principles for Creating Web Sites / Edition 3, publ. Jan. 15, 2009, Yale University Press. http://webstyleguide.com/wsg3/, last visited Jan. 17, 2017.

Rejection under 35 U.S.C. § 103

The Examiner relies on DeHaan for disclosing common components across two different software products and includes generating a dependency map as shown in Figure 1 and noted in paragraph 20 (Ans. 4; Final Act. 7–8).

Appellants contend although there is a dependency map mentioned in DeHaan (¶ 20), there is nothing in paragraph 20 about the dependency map based on candidate subtrees as claimed (App. Br. 14). The Examiner's Answer appears to rely on Gupta for the elements not found in DeHaan in the Final Action (Ans. 4).

As we find Gupta does not disclose what it was relied on for, and as DeHaan fails to cure the deficiencies of Gupta, we agree with Appellants the Examiner erred in finding claims 1–9 and 12–14 obvious over the collective teachings of the cited references.

DECISION

The Examiner's decision rejecting claims 1–10 and 12–14 is reversed.

REVERSED